SubTST: A Combination of Sub-word Latent Topics and Sentence Transformer for Semantic Similarity Detection

Binh Dang[1], Tran-Thai Dang[2] and Le-Minh Nguyen[1]
[1]*Japan Advanced Institute of Science and Technology, Nomi, Ishikawa, Japan*
[2]*Vingroup Big Data Institute, Hanoi, Vietnam*

Keywords: Sub-word, Topic Modeling, Sentence Transformer, Bi-encoder, Semantic Similarity Detection.

Abstract: Topic information has been useful for semantic similarity detection. In this paper, we present a study on a novel and efficient method to incorporate the topic information with Transformer-based models, which is called the Sub-word Latent Topic and Sentence Transformer (SubTST). The proposed model basically inherits the advantages of the SBERT (Reimers and Gurevych, 2019) architecture, and learns latent topics in the sub-word level instead of the document or word levels as previous work. The experimental results illustrate the effectiveness of our proposed method that significantly outperforms the SBERT, and the tBERT (Peinelt et al., 2020), two state-of-the-art methods for semantic textual detection, on most of the benchmark datasets.

1 INTRODUCTION

The Semantic Textual Similarity Detection (STS) is a crucial task in the Natural Language Understanding (NLU). This aims to determine the semantic correlation between a pair of sentences. STS was applied in a lot of NLP application ranging from information retrieval to paraphrase detection. An example of the STS is shown in Table 1.

So far, there are various studies on the STS. Wu et al. (Wu et al., 2017) utilize various features (e.g., words, topics, lexical features, etc) with the convolution neural network (CNN) for classifying pairs of sentences in the SemEval-2017 Task 3 CQA competition. In (Tan et al., 2018), four attention functions are used to match sentence pairs under the matching-aggregation framework. The SBERT (Reimers and Gurevych, 2019) combines the pre-trained contextual BERT (Devlin et al., 2019) network and Siamese network (Schroff et al., 2015), which achieved a significant improvement on this task.

The effectiveness by using latent topic information as features for information retrieval, recommendation system, and STS has been pointed out in several studies (Qin et al., 2009), (Ovsjanikov and Chen, 2010), (Tran et al., 2015), (Dang et al., 2020) and (Wu et al., 2017). In addition, the topic embedding is concatenated with the sentence embedding produced by the BERT, which is called tBERT (Peinelt et al., 2020). This helps to improve the BERT's performance for STS and illustrates the efficacy of topic information as well.

Table 1: Example of semantic similarity detection.

Sentence pair	Similar /non-similar
How do I get funding for my web based startup idea ? How do I get seed funding pre product ?	similar
What is ecstasy ? How addictive is ecstasy ?	non - similar

In this work, we propose a novel method for enhancing the capacity of Transformer-based models for the STS by incorporating topic information over sub-words. The method is called the Sub-word Latent Topic and Sentence Transformer (SubTST). The major contributions of our work are as follows:

- The SubTST essentially uses the SBERT architecture (a.k.a., bi-encoder). In this method, the latent topics are learned over sub-words instead of documents/words as in previous work. In addition, we transform a concatenation of output vectors generated by the topic model and the Transformer-based model by a transfer layer. Our approach is essentially different from the tBERT's approach (Peinelt et al., 2020) - a state-of-the-art method of using the topic information with

Dang, B., Dang, T. and Nguyen, L.
SubTST: A Combination of Sub-word Latent Topics and Sentence Transformer for Semantic Similarity Detection.
DOI: 10.5220/0010775100003116
In *Proceedings of the 14th International Conference on Agents and Artificial Intelligence (ICAART 2022) - Volume 3*, pages 91-97
ISBN: 978-989-758-547-0; ISSN: 2184-433X

BERT for improving the STS performance. In the tBERT, a sentence pair is concatenated by adding the CLS token before being embedded by the BERT encoder (a.k.a., cross-encoder). The document/word topic embedding vector is concatenated with the output of BERT encoder before being fed to a softmax classification layer.

- We demonstrate that the proposed model prominently outperforms two state-of-the-art methods for semantic textual detection, the SBERT and tBERT, on most of the benchmark datasets (section 3).

2 SUB-WORD LATENT TOPIC AND SENTENCE TRANSFORMER (SUBTST)

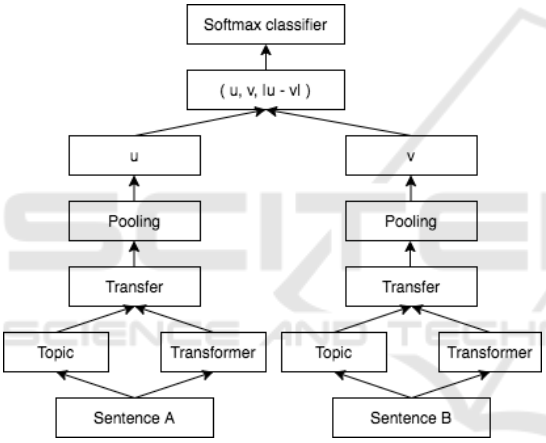

Figure 1: SubTST architecture with softmax objective function.

The key idea in our work is to unify the utilization of the lexicon unit for learning both topic-based and BERT-based sentence embedding vectors. That means both topic-based and BERT-based vectors are drawn from the same distribution of lexicon units. Hence, we learn latent topics at the sub-word level and use such information with the bi-encoder architecture for semantic similarity detection. Figure 1 illustrates the architecture of our proposed model.

Given a pair of sentences (sentences A and B), we encode each sentence individually by using both topic model and Transformer-based model. In the topic-based representation, each sentence s is characterized by a topic-term matrix of size $k \times N_s$, denoted by M_t where k is the number of latent topics and N_s is the number of sub-words in each sentence:

$$M_t = TopicModel(s) \in R^{k \times N_s} \quad (1)$$

The LDA (Blei et al., 2003) is a popular topic model to learn latent topics. In this work, we use the version of LDA presented in (Hoffman et al., 2010). We denote the output of pre-trained Transformer-based models given a sentence s by $M_c \in R^{m \times N_s}$ where m is the internal hidden size of transformer model. In the SubTST, we utilize the BERT_{base} model for encoding sentences.

$$M_c = Transformer(sentence) \in R^{m \times N_s} \quad (2)$$

To aggregate the topic information with the output of Transformer-based models, we concatenate M_c and M_t into M_{ct} as the following:

$$M_{ct} = \begin{pmatrix} M_c \\ M_t \end{pmatrix} \in R^{(m+k) \times N_s} \quad (3)$$

then feed M_{ct} to a transfer layer as illustrated in Figure 2. The transfer layer is constructed by a Feed-forward network with the Dropout and Layer Normalization:

$$\begin{aligned} h &= WM_{ct} + B \\ h &= LayerNorm(Dropout(h)) \end{aligned} \quad (4)$$

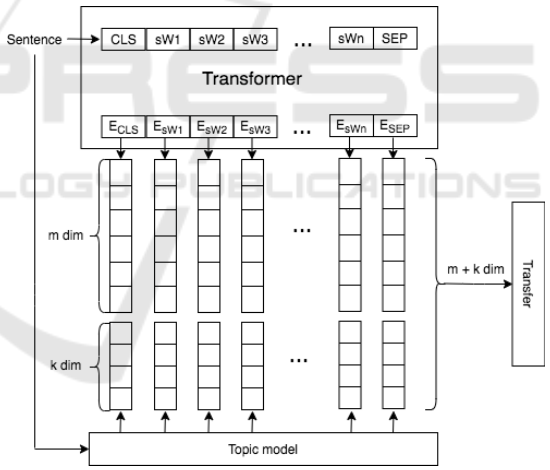

Figure 2: Combination of topic embedding and word embedding.

Similarly to the SBERT (Reimers and Gurevych, 2019), we add the pooling layer that helps to combine sub-word's vectors into an unique sentence's vector. In our work, we experiment two pooling strategies that are the Mean-strategy and Max-strategy. Given $h \in R^{(m+k) \times N_s}$, the embedding of sentence $u \in R^{m+k}$ is estimated by:

- Mean-strategy:

$$u = MEAN(h) \quad (5)$$

- Max-strategy:

$$u = MAX(h) \tag{6}$$

Outputs of two pooling layers, u, v are diffused by additionally estimating $|u-v|$, then a concatenation of u, v, and $|u-v|$ is fed to a softmax classification layer (Bishop, 2006):

$$O = softmax(W_t(u, v, |u-v|)) \quad with\ W_t \in R^{3\times(m+k)\times l} \tag{7}$$

where $m+k$ is dimension of sentence embedding; l is size of set labels.

3 EVALUATION

3.1 Experimental Setup

We evaluate the performance of SubTST on the semantic similarity detection by using three datasets Quora, MSRP, and SemEval CQA with statistics information shown in Table 2.

Quora: The duplicated questions dataset containing over 400,000 question pairs with two labels (duplicate:1 or non-duplicate: 0). The task is defined as same as a classification task. The train/dev/test sets are divided according to (Wang et al., 2017).

MSRP: The Microsoft Research Paraphrase (Dolan and Brockett, 2005) dataset that consists of over 5000 sentence pairs collected from news. Each pair has an annotation indicating paraphrased or non-paraphrased pair.

SemEval CQA:[1] (Nakov et al., 2015),(Nakov et al., 2016),(Nakov et al., 2017) The SemEval CQA is the combination between 3 subtasks A, B, and C base on questions and answers on Qatar Living forum. In this paper, we only evaluate subtask A and subtask B. Subtask A(Question-Comment Similarity) give a question and its first 10 comments in the question thread, rerank these 10 comments according to their relevance with respect to the question. Subtask B(Question-Question Similarity) gives a new question and the set of the first 10 related questions, rerank the related questions according to their similarity with respect to the original question. However, both of them are organized as a classification rather than a ranking problem. Each question-comment pair in subtask A may be in one of the labels("Bad"/"PotentiallyUsefu": 0, "Good": 1). In subtask B, set label for question-question pair are "Irrelevant": 0 and "Relevant"/"PerfectMatch": 1. Subtask C (Question-External Comment Similarity) is the same subtask A: with a new question, 100 comments (each related question in subtask B has 10 comments) evaluate their relevance concerning the original question.

Table 2: The information of benchmark datasets.

Dataset	#length	#samples	#topics
Quora	13	404000	90
MSRP	22	5000	80
SemEval CQA (A)	48	26000	70
SemEval CQA (B)	52	4000	80
SemEval CQA (C)	45	47000	70

We quantitatively compare the SubTST with several previous systems for semantic similarity detection based on the accuracy and F1-score. The baselines include: (i) SBERT[2]; (ii) tBERT; (iii) SwissAlps (Deriu and Cieliebak, 2017) - a method is built on a siamese CNN architecture and evaluated on the SemEval CQA; (iv) KeLP (Filice et al., 2017) - a refinement of the kernel-based sentence pair modeling. In which, the SBERT and tBERT are built based on the pre-trained $BERT_{base}$ model.

As mentioned in section 2, each input sentence is encoded by the $BERT_{base}$, and the LDA is used for learning latent topics, which is better than other topic models such as GSDMM (Yin and Wang, 2014) as mentioned in the study on tBERT (Peinelt et al., 2020). The best number of topics for each dataset is considered based on an analysis of the tBERT. The SubTST uses as same number of topics as the tBERT (mentioned in Table 2). Regarding the pooling layer, we apply both Max-strategy and Mean-strategy in our experiments. We setup two configurations for the topic-based sentence representation, one is that the tensor of topic embedding is learnable (denoted by SubTST-mean-train topic, SubTST-max-train topic), the other is that such a tensor is frozen (denoted by SubTST-mean, SubTST-max).

3.2 Experimental Results

We make a comparison between the proposed method and baseline systems that is shown in Table 3.

Overall, the SubTST significantly outperforms baseline systems (including state-of-the-art methods such as SBERT and tBERT) in most of the benchmark datasets. The experimental results prominently show the effectiveness of SubTST. In addition, we observe that the mean strategy is more appropriate for semantic similarity detection than the max strategy because it often gives a better performance than the max strategy.

[1] https://alt.qcri.org/semeval2017/task3/

[2] https://github.com/UKPLab/sentence-transformers

Table 3: Results of methods on datasets: MSRP, Quora, SemEval based on Accuracy and F1-score (use BERT$_{base}$). *: the results that reported in paper.

	Accuracy					F1				
	MSRP	Quora	SemEval subtask A	SemEval subtask B	SemEval subtask C	MSRP	Quora	SemEval subtask A	SemEval subtask B	SemEval subtask C
Previous system										
SwissAlps*	-	-	-	-	-	-	-	-	43.3	-
KeLP*	-	-	-	-	-	-	-	69.87	50.64	-
tBERT*	-	-	-	-	-	**88.4**	90.5	76.8	52.4	27.3
Our implementation										
SBERT - mean	71.6	89.9	76.7	62.2	67.0	80.9	89.9	76.9	47.9	32.14
SBERT - max	69.7	88.7	77.3	60.2	66.7	80.1	88.4	77.0	33.9	31.89
SubTST - mean	69.6	89.9	76.6	64.8	67.1	79.0	90.1	76.5	**61.2**	32.28
SubTST - max	71.4	89.2	77.5	61.4	67.0	80.9	89.1	77.7	44.7	32.22
SubTST - mean - train topic	**72.2**	**90.5**	**78.2**	**69.2**	**67.3**	**82.3**	**90.7**	**77.8**	54.2	**32.58**
SubTST - max - train topic	70.8	89.2	77.5	61.1	66.8	81.1	89.0	77.2	45.7	32.04

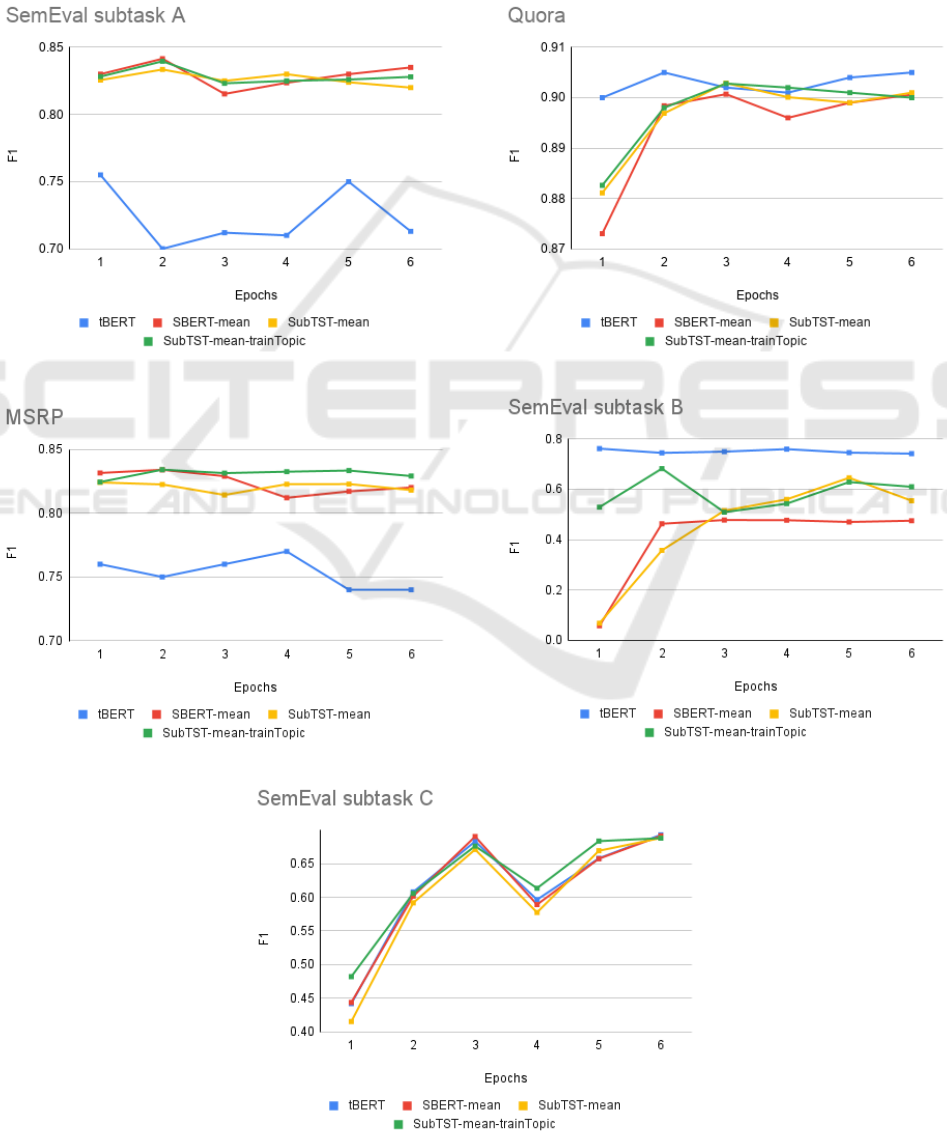

Figure 3: Performance of SubTST and baselines on dev set on datasets.

To verify the likelihood of our hypothesis on the efficacy of using sub-word latent topics, we compare the SubTST with/without finetuning topic-subword matrix with the tBERT. The experimental results in Table 3 show that: (i) for the SemEval subtasks A, B datasets the SubTST with both configurations for the topic-subword matrix results in better F1 score than the tBERT; (ii) for the Quora dataset, the SubTST with frozen topic-based embedding is competitive towards the tBERT, and our model performs better than tBERT when setting the learnable topic-based embedding in the training process. In conclusion, these empirical experiments demonstrate the effectiveness of combining sub-word topic information with Transformer-based models for semantic similarity detection.

When comparing with tBERT on the MSRP dataset, our proposed method is worse. In fact, the MSRP dataset contains a small number of samples with a lot of named entities, which poses a big challenge for classification systems. As mentioned in (Humeau et al., 2020), the cross-encoder is often better than the bi-encoder due to the full self-attention mechanism, however, the cross-encoder is too low in practical use. Hence, for such data, the tBERT has more advantages than our proposed method due to using the cross-encoder architecture. For other datasets, in spite of using the bi-encoder architecture that is weaker than the cross-encoder one, the SubTST outperforms the tBERT. Therefore, this demonstrates the power of incorporating latent topics learned from sub-words.

In the evaluation on Subtask B, the SubTST with frozen topic-based embedding is better than the SubTST with learn topic-based embedding (trainable topic embeddings - 54.2 F1 and their frozen - 61.2 F1). We can understand this phenomenon. For SemEval subtask B, this task is to compare the semantics similarity between question and question. The length of each sentence in a pair is often too long. This is a special characteristic of subtask B. We think that it is the reason for the difference.

Figure 3 depicts our analysis of the training process through each epoch based on the development set. We visualize the fine-tuning process of the SubTST with tBERT and SBERT over 6 epochs. We found that the SubTST achieves the peak of the F1 score after 1 or 2 epochs, in the next epochs, the change of the F1 score tends to be monotonic with a small amplitude in comparison with tBERT. Hence, this shows the stability in the training SubTST which is basically resulted from unifying the lexicon unit for topic models and Transformer-based models.

3.3 Discussion

As experiments, we can see that the power of a transformer model based bi-encoder with topic information based on sub-words. In some previous researches as SBERT(Reimers and Gurevych, 2019), the authors detailed that the complexity for finding the foremost comparable sentence match in a collection of 10,000 sentences is diminished from 65 hours with BERT to the computation of 10,000 sentence embedding (about 5 seconds with SBERT) and computing cosine likeness (about 0.01 seconds). This is proof of the ability of the bi-encoder as SBERT or SubTST when applying in real.

Normally, it's easy to understand the practical meaning of the latent topics over words/documents. However, topic modeling on sub-words can bring a lot of benefits instead of words/document: (*i*) The model can reduce the number of unknown words (the out of vocab words) in the usage process. When using a topic model, the vocab of the topic model often fits with the corpus. So when applying for another corpus, the number of unknown words could be very large . Using latent topics over sub-words sure significantly reduces "out of vocab"; (*ii*) With transformer-based models such as BERT, sub-words are the base unit of a sentence when processing. To easily combine topic models and transformer-based models, we decided to use sub-words for topic models.

4 CONCLUSION

This paper presents a new method for incorporating latent topic information with Transformer-based models, called the SubTST. This method aims to unify the lexicon unit for both the topic model and the Transformer-based model. The experimental results show that our model outperforms all baseline models including state-of-the-art models in semantic similarity detection. Hence, this indicates the effectiveness of our proposed method. In addition, the SubTST is built based on the bi-encoder architecture, so it has more advantages in practical applications (e.g., inference time) in comparison with tBERT. Moreover, the fine-tuning process of SubTST is fairly fast to reach the peak performance, and stable. Our work also reveals the effectiveness of unifying the data distribution (a.k.a, using the same lexicon level) for learning topic-based and Transformer-based sentence representations, which will support further studies on this field.

ACKNOWLEDGEMENTS

This work was supported by JSPS Kakenhi Grant Number 20H04295, 20K20406, and 20K20625.

REFERENCES

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag.

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022.

Dang, T.-B., Nguyen, H.-T., and Nguyen, L.-M. (2020). Latent topic refinement based on distance metric learning and semantics-assisted non-negative matrix factorization. In *Proceedings of the 34th Pacific Asia Conference on Language, Information and Computation*, pages 70–75, Hanoi, Vietnam. Association for Computational Linguistics.

Deriu, J. M. and Cieliebak, M. (2017). SwissAlps at SemEval-2017 task 3: Attention-based convolutional neural network for community question answering. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 334–338. Association for Computational Linguistics.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Dolan, W. B. and Brockett, C. (2005). Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing (IWP2005)*.

Filice, S., Da San Martino, G., and Moschitti, A. (2017). KeLP at SemEval-2017 task 3: Learning pairwise patterns in community question answering. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 326–333. Association for Computational Linguistics.

Hoffman, M., Bach, F., and Blei, D. (2010). Online learning for latent dirichlet allocation. In *Advances in Neural Information Processing Systems*, volume 23. Curran Associates, Inc.

Humeau, S., Shuster, K., Lachaux, M.-A., and Weston, J. (2020). Poly-encoders: Architectures and pre-training strategies for fast and accurate multi-sentence scoring. In *International Conference on Learning Representations*.

Nakov, P., Hoogeveen, D., Màrquez, L., Moschitti, A., Mubarak, H., Baldwin, T., and Verspoor, K. (2017). SemEval-2017 task 3: Community question answering. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 27–48. Association for Computational Linguistics.

Nakov, P., Màrquez, L., Magdy, W., Moschitti, A., Glass, J., and Randeree, B. (2015). SemEval-2015 task 3: Answer selection in community question answering. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 269–281. Association for Computational Linguistics.

Nakov, P., Màrquez, L., Moschitti, A., Magdy, W., Mubarak, H., Freihat, A. A., Glass, J., and Randeree, B. (2016). SemEval-2016 task 3: Community question answering. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 525–545. Association for Computational Linguistics.

Ovsjanikov, M. and Chen, Y. (2010). Topic modeling for personalized recommendation of volatile items. In *Machine Learning and Knowledge Discovery in Databases*, pages 483–498. Springer Berlin Heidelberg.

Peinelt, N., Nguyen, D., and Liakata, M. (2020). tBERT: Topic models and BERT joining forces for semantic similarity detection. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7047–7055, Online. Association for Computational Linguistics.

Qin, Z., Thint, M., and Huang, Z. (2009). Ranking answers by hierarchical topic models. In *Next-Generation Applied Intelligence*, pages 103–112. Springer Berlin Heidelberg.

Reimers, N. and Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992. Association for Computational Linguistics.

Schroff, F., Kalenichenko, D., and Philbin, J. (2015). Facenet: A unified embedding for face recognition and clustering. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 815–823.

Tan, C., Wei, F., Wang, W., Lv, W., and Zhou, M. (2018). Multiway attention networks for modeling sentence pairs. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pages 4411–4417. International Joint Conferences on Artificial Intelligence Organization.

Tran, Q. H., Tran, V. D., Vu, T. T., Nguyen, M. L., and Pham, S. B. (2015). JAIST: Combining multiple features for answer selection in community question answering. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 215–219, Denver, Colorado. Association for Computational Linguistics.

Wang, Z., Hamza, W., and Florian, R. (2017). Bilateral multi-perspective matching for natural language sentences. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 4144–4150.

Wu, G., Sheng, Y., Lan, M., and Wu, Y. (2017). ECNU at SemEval-2017 task 3: Using traditional and deep

learning methods to address community question answering task. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 365–369, Vancouver, Canada. Association for Computational Linguistics.

Yin, J. and Wang, J. (2014). A dirichlet multinomial mixture model-based approach for short text clustering. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, page 233–242, New York, NY, USA. Association for Computing Machinery.